

AD-A060 674

ARMY ARMAMENT RESEARCH AND DEVELOPMENT COMMAND ABERD--ETC F/G 9/2  
THE DAVE SYSTEM: A CRITIQUE AND GUIDE FOR USE AT THE BALLISTIC --ETC(U)  
AUG 78 M A HIRSCHBERG, J LACETERA

UNCLASSIFIED

ARBRL-MR-02858

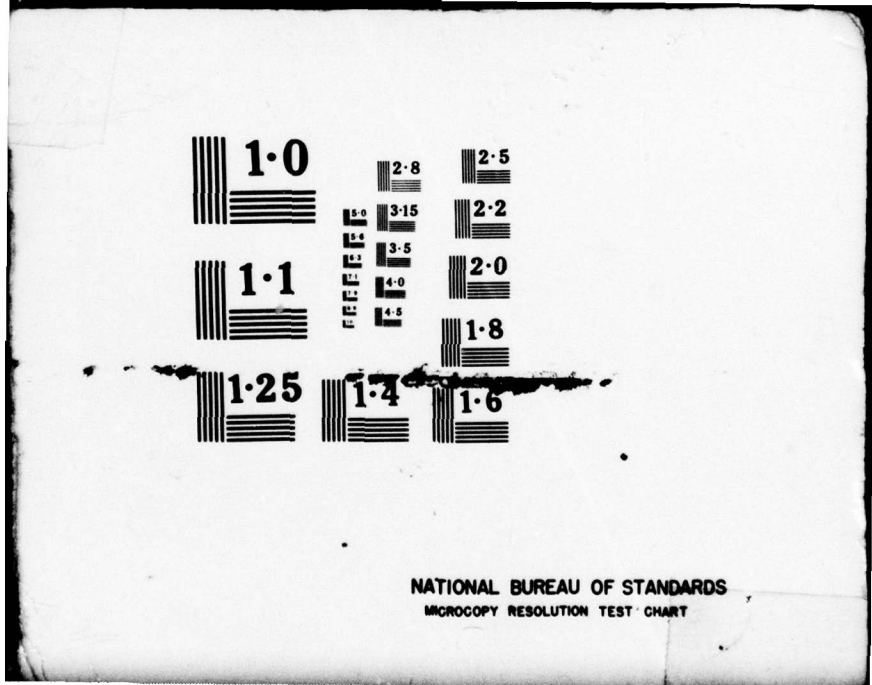
SBIE-AD-E430 107

NL

1 OF 1  
ADA  
060674



END  
DATE  
FILMED  
1 -79  
DDC



12  
b.s.

**LEVEL II**

AD-E430107

AD A 0 6 0 6 7 4

MEMORANDUM REPORT ARBRL-MR-02858

THE DAVE SYSTEM: A CRITIQUE AND GUIDE FOR  
USE AT THE BALLISTIC RESEARCH LABORATORY

Morton A. Hirschberg  
Joseph Lacetera  
William Buchheister

DDC FILE COPY

August 1978

DDC  
RECEIVED  
NOV 1 1978  
B



US ARMY ARMAMENT RESEARCH AND DEVELOPMENT COMMAND  
BALLISTIC RESEARCH LABORATORY ✓  
ABERDEEN PROVING GROUND, MARYLAND

Approved for public release; distribution unlimited.

78 09 18 067

Destroy this report when it is no longer needed.  
Do not return it to the originator.

Secondary distribution of this report by originating  
or sponsoring activity is prohibited.

Additional copies of this report may be obtained  
from the National Technical Information Service,  
U.S. Department of Commerce, Springfield, Virginia  
22161.

The findings in this report are not to be construed as  
an official Department of the Army position, unless  
so designated by other authorized documents.

*The use of trade names or manufacturers' names in this report  
does not constitute indorsement of any commercial product.*



TABLE OF CONTENTS

|   | Page |
|---|------|
| I. Introduction . . . . .   | 5    |
| II. DAVE and Critique of DAVE . . . . .   | 6    |
| III. Instructions on the use of DAVE at the Ballistic<br>Research Laboratory. . . . . | 7    |
| IV. Summary. . . . .  | 8    |
| REFERENCES . . . . .  | 11   |
| APPENDIX A: FORTRAN Compilation of Test Program<br>Which Shows No Errors . . . . .    | 13   |
| APPENDIX B: Results of DAVE Execution on<br>Test Program . . . . .                    | 17   |
| DISTRIBUTION LIST . . . . .   | 35   |

|                                 |               |                                     |
|---------------------------------|---------------|-------------------------------------|
| ACCESSION for                   |               |                                     |
| WTIS                            | White Section | <input checked="" type="checkbox"/> |
| DDC                             | Buff Section  | <input type="checkbox"/>            |
| UNANNOUNCED                     |               | <input type="checkbox"/>            |
| JUSTIFICATION _____             |               |                                     |
| BY _____                        |               |                                     |
| DISTRIBUTION/AVAILABILITY CODES |               |                                     |
| Dist.                           | AVAIL.        | SPECIAL                             |
| A                               |               |                                     |

PRECEDING PAGE BLANK-NOT FILMED

## I. INTRODUCTION

Dave<sup>1</sup> is one of several program validation techniques in use for the analysis of FORTRAN programs. The need for techniques which enable programmers to validate programs and reduce costs is well documented<sup>2,3</sup>.

DAVE is a static analyzer. That is, DAVE looks at FORTRAN programs and provides error analysis and documentation of code without executing the program. While this technique is not foolproof, it presents the user considerable analysis of FORTRAN programs above that provided by FORTRAN compilers. A comparison of a "clean" FORTRAN program compiled on a CDC Cyber 173 and analyzed by DAVE is shown in the appendices (Appendix A and B).

Fairley<sup>4</sup>, in a recent article, listed ten items of information which can be obtained by static analysis. These are quoted verbatim below:

*"The information that can be obtained by static analysis includes (1) syntactic error messages; (2) number of occurrences of source statements by type; (3) cross-reference maps of identifier usage; (4) analysis of how the identifiers are used in each statement (data source, data sink, calling parameter, dummy parameter, subscript, etc.); (5) subroutines and functions called by each routine; (6) uninitialized variables; (7) variables set but not used; (8) isolated code segments that cannot be executed under any set of input data; (9) departures from coding standards (both language standards and local practice standards); and (10) misuses of global variables, common variables, and parameter lists (incorrect number of parameters, mismatched types, uninitialized input parameters, output parameters not assigned to, output parameters assigned to but never used, parameters never used for either input or output, etc.)."*

<sup>1</sup>Osterweil, L. J., Fosdick, L. D., "DAVE - A Validation, Error Detection and Documentation System for FORTRAN Programs", Department of Computer Science, University of Colorado, TR #CU-CS-071-75, February 1975.

<sup>2</sup>Elspar, B., Levitt, K. N., Waldinger, J. and Waksman, A., "An Assessment of Techniques for Proving Program Correctness", ACM Computing Survey, 4, pp. 97-147, June 1972.

<sup>3</sup>Boehm, B., "Software and Its Impact: A Quantitative Assessment", Data-mation, 14, 5, pp 48-59, May 1973.

<sup>4</sup>Fairley, R. E., "Tutorial: Static Analysis and Dynamic Testing of Computer Software", Computer, pp. 14-23, April 1978.

Prior to using DAVE, one author had some experience with RXVP<sup>5</sup>. Outside of that, the authors had no prior experience with validating techniques other than that provided by standard FORTRAN compilers. We feel, therefore, that we can present an unbiased evaluation of DAVE.

## II. DAVE AND CRITIQUE OF DAVE

The DAVE system is well documented in Reference 1; however, a few of its characteristics and highlights will be presented.

DAVE is written in ANSI FORTRAN and contains approximately 20,000 source statements. DAVE consists of a number of files which can be and have been combined to form a procedure (see Section III). The most important of these files for the user is the FORTRAN program to be analyzed. This may consist of a single subprogram, a group of subprograms, or a main program and its subprograms.

DAVE attempts to locate violations of FORTRAN rules and thus detect the presence of common programming errors. DAVE produces 13 error messages (e.g., the number of arguments in the parameter lists do not correspond), 24 warning messages (e.g., a variable in a parameter list is used for neither input nor output), and 5 general messages (e.g., subprogram "NAME" is never called). The output from DAVE is extensive and a good portion of it is informative in nature and points to "errors" that may not affect the running of the program. Programmers who use sophisticated techniques (not all of which are allowed under all versions of FORTRAN compilers), may find a good number of warning errors after putting their programs through DAVE.

The authors have taken the simple input case provided with DAVE and cleaned up all the errors found by the CDC NOSBE FORTRAN compiler. These "clean" programs were then processed through DAVE which found numerous errors; errors serious enough to show that the programs were indeed not valid. Portions of these results are shown in the appendices.

As a tool, DAVE does perform many of the ten tasks outlined by Fairley. It does not provide a complete set of syntactic error messages but does provide many such messages. DAVE does not provide the number of occurrences of source statements by type, cross reference maps of identifier usage, or an analysis of how identifiers are used. DAVE does provide information as to all the other items listed by Fairley.

Finally, we feel that the limits of application of DAVE to FORTRAN programs should be mentioned here; and these are quoted verbatim below:

---

<sup>5</sup>Miller, E. F., Jr., "RXVP: An Automated Verification System for FORTRAN", in Proc. Computer Science and Statistics: 8th Annual Symposium on the Interface, Los Angeles, CA, February 1975.

*"The limits on the size of the FORTRAN programs to be analyzed by DAVE are: a maximum of 100 subprograms run through together, each of which has no more than 500 blocks. All declarations are counted as one block; otherwise each statement is a block, with logical IF's counting as two blocks; COMMENTS and Formats are excluded from the count. Although the limit is 100 program units run together, the larger the group, the more strain placed on all internal arrays and overflow may occur. Detailed information on all size limitations will appear in the User Manual<sup>6</sup>".*

Although our experience with DAVE is still limited, we hope that this document will encourage the use of DAVE so that we can gather further statistics and report upon these in the future.

### III. INSTRUCTIONS ON THE USE OF DAVE AT THE BALLISTIC RESEARCH LABORATORY

This section deals with the mechanics of using DAVE; that is, which procedures must be used, what file DAVE resides on, and what the run stream for DAVE is.

For the convenience of users at the BRL we have set up a procedure file to execute the DAVE code. The following is a sample deck for its use:

```
BRL76, STMFZ,T40.  
ACCOUNT (BRL76)  
ATTACH (OLDPL, PLTESTFLC, ID = CMCWB)  
UPDATE (F, C=TEST, D,8)  
BEGIN, DAVE, DAVE, INPUT, TEST.  
7/8/9  
7/8/9  
SI=ON  
7/8/9
```

The ATTACH command accesses the permanent file PLTESTFLC and assigns it the local file name OLDPL. PLTESTFLC is an UPDATE program library which we wish to use as input to DAVE, and can have any name. In this particular example it contained our modified version of the test program which was provided with DAVE to exercise its capabilities. As noted earlier, this program compiled without errors on the 4.6(OPT=2)compiler.

The UPDATE command causes UPDATE to write the compile output decks on the file named TEST. It is a full update and the output has 80 data columns. This file is to be used as the input file by the DAVE procedure file, and must have the same name on both cards.

<sup>6</sup>"Installing DAVE on a Computer", University of Colorado, Undated Notes.

The FORTRAN program to be analyzed may consist of a single subprogram, several subprograms, a main program and subprograms, or just a main program.

Finally, the BEGIN card causes execution of the DAVE procedure file. Note that the file INPUT contains the record SI=ON which indicates that simulations of missing subprograms are desired. Another input option to DAVE uses the keyword SU which can be used to suppress the printing of errors, warnings, or messages.

If the test program were not on an UPDATE program library but on cards, it would have to follow the options cards in the input file, and be separated from them by an end-of-file.

It should be noted that the dayfile in Appendix B shows the entire job control stream, generated by the "PROC", and that these instructions are not part of the job control deck.

Appendix A contains the listing of the FORTRAN compilation of the main program on the permanent file PLTESTFLC. It is free of error messages or diagnostic information of any type, as were all of the subroutines (not shown). Appendix B contains the output resulting from executing DAVE on this same main program. The output is extensive showing errors, warnings, and diagnostic messages. It is representative of the output resulting from DAVE operating on the subroutines. From this output it is apparent that one obvious aspect of DAVE's superiority over FORTRAN compilers is its ability to analyze coding in the context of subroutines and functions called by a program or subprograms.

#### IV. SUMMARY

Based on limited use, DAVE does everything it is purported to do. From the standpoint of economics, DAVE is inexpensive to purchase, and in an era of expanding work and a decreasing work force, a worthwhile error detector.

Other new tools being developed for the analysis of FORTRAN programs are either not yet available for general consumption<sup>7</sup>, or require additional resources for use<sup>8</sup>. In the latter case, not everyone can afford

<sup>7</sup>Clarke, L., "A System to Generate Test Data and Symbolically Execute Programs", Department of Computer Science, University of Colorado, #CU-CS-060-75, February 1975.

<sup>8</sup>Browne, J. C. and Johnson, D. B., "FAST: A Second Generation Program Analysis System", Proceedings of 3rd International Conference on Software Engineering, IEEE Catalog No. 78CH1317-7C, pp. 142-148, May 1978.

the lease or purchase of a data base analysis system (in one case, System 2000). In the former case, we are awaiting the use of a symbolic evaluator.

## REFERENCES

1. Osterweil, L. J., Fosdick, L. D., "DAVE - A Validation, Error Detection and Documentation System for FORTRAN Programs", Department of Computer Science, University of Colorado, TR #CU-CS-071-75, February 1975.
2. Elspas, B., Levitt, K. N., Waldinger, J. and Waksman, A., "An Assessment of Techniques for Proving Program Correctness", ACM Computing Survey, 4, pp. 97-147, June 1972.
3. Boehm, B., "Software and Its Impact: A Quantitative Assessment", Datamation, 14, 5, pp. 48-59, May 1973.
4. Fairley, R. E., "Tutorial: Static Analysis and Dynamic Testing of Computer Software", Computer, pp. 14-23, April 1978.
5. Miller, E. F., Jr., "RXVP: An Automated Verification System for FORTRAN", in Proc. Computer Science and Statistics: 8th Annual Symposium on the Interface, Los Angeles, CA, February 1975.
6. "Installing DAVE on a Computer", University of Colorado, Undated Notes.
7. Clarke, L., "A System to Generate Test Data and Symbolically Execute Programs", Department of Computer Science, University of Colorado, "CU-CS-060-75, February 1975.
8. Browne, J. C. and Johnson, D. B., "FAST: A Second Generation Program Analysis System", Proceedings of 3rd International Conference on Software Engineering", IEEE Catalog No. 78CH1317-7C, pp. 142-148, May 1978.

APPENDIX A

FORTRAN COMPILATION OF TEST PROGRAM  
WHICH SHOWS NO ERRORS.

```

      DIMENSION A(10), B(10), C(10), D(10), E(10), F(10), G(10), H(10), I(10), J(10)
      DIMENSION K(10), L(10), M(10), N(10), O(10), P(10), Q(10), R(10), S(10), T(10)
      DIMENSION U(10), V(10), W(10), X(10), Y(10), Z(10), AA(10), AB(10), AC(10), AD(10)
      DIMENSION AE(10), AF(10), AG(10), AH(10), AI(10), AJ(10), AK(10), AL(10), AM(10), AN(10)
      DIMENSION AO(10), AP(10), AQ(10), AR(10), AS(10), AT(10), AU(10), AV(10), AW(10), AX(10)
      DIMENSION AY(10), AZ(10), BA(10), BB(10), BC(10), BD(10), BE(10), BF(10), BG(10), BH(10)
      DIMENSION BI(10), BJ(10), BK(10), BL(10), BM(10), BN(10), BO(10), BP(10), BQ(10), BR(10)
      DIMENSION BS(10), BT(10), BU(10), BV(10), BW(10), BX(10), BY(10), BZ(10), CA(10), CB(10)
      DIMENSION CC(10), CD(10), CE(10), CF(10), CG(10), CH(10), CI(10), CJ(10), CK(10), CL(10)
      DIMENSION CM(10), CN(10), CO(10), CP(10), CQ(10), CR(10), CS(10), CT(10), CU(10), CV(10)
      DIMENSION CW(10), CX(10), CY(10), CZ(10), DA(10), DB(10), DC(10), DD(10), DE(10), DF(10)
      DIMENSION DG(10), DH(10), DI(10), DJ(10), DK(10), DL(10), DM(10), DN(10), DO(10), DP(10)
      DIMENSION DQ(10), DR(10), DS(10), DT(10), DU(10), DV(10), DW(10), DX(10), DY(10), DZ(10)
      DIMENSION EA(10), EB(10), EC(10), ED(10), EE(10), EF(10), EG(10), EH(10), EI(10), EJ(10)
      DIMENSION EK(10), EL(10), EM(10), EN(10), EO(10), EP(10), EQ(10), ER(10), ES(10), ET(10)
      DIMENSION EU(10), EV(10), EW(10), EX(10), EY(10), EZ(10), FA(10), FB(10), FC(10), FD(10)
      DIMENSION FE(10), FF(10), FG(10), FH(10), FI(10), FJ(10), FK(10), FL(10), FM(10), FN(10)
      DIMENSION FO(10), FP(10), FQ(10), FR(10), FS(10), FT(10), FU(10), FV(10), FW(10), FX(10)
      DIMENSION FY(10), FZ(10), GA(10), GB(10), GC(10), GD(10), GE(10), GF(10), GG(10), GH(10)
      DIMENSION GI(10), GJ(10), GK(10), GL(10), GM(10), GN(10), GO(10), GP(10), GQ(10), GR(10)
      DIMENSION GS(10), GT(10), GU(10), GV(10), GW(10), GX(10), GY(10), GZ(10), HA(10), HB(10)
      DIMENSION HC(10), HD(10), HE(10), HF(10), HG(10), HH(10), HI(10), HJ(10), HK(10), HL(10)
      DIMENSION HM(10), HN(10), HO(10), HP(10), HQ(10), HR(10), HS(10), HT(10), HU(10), HV(10)
      DIMENSION HW(10), HX(10), HY(10), HZ(10), IA(10), IB(10), IC(10), ID(10), IE(10), IF(10)
      DIMENSION IG(10), IH(10), II(10), IJ(10), IK(10), IL(10), IM(10), IN(10), IO(10), IP(10)
      DIMENSION IQ(10), IR(10), IS(10), IT(10), IU(10), IV(10), IW(10), IX(10), IY(10), IZ(10)
      DIMENSION JA(10), JB(10), JC(10), JD(10), JE(10), JF(10), JG(10), JH(10), JI(10), JJ(10)
      DIMENSION JK(10), JL(10), JM(10), JN(10), JO(10), JP(10), JQ(10), JR(10), JS(10), JT(10)
      DIMENSION JU(10), JV(10), JW(10), JX(10), JY(10), JZ(10), KA(10), KB(10), KC(10), KD(10)
      DIMENSION KE(10), KF(10), KG(10), KH(10), KI(10), KJ(10), KK(10), KL(10), KM(10), KN(10)
      DIMENSION KO(10), KP(10), KQ(10), KR(10), KS(10), KT(10), KU(10), KV(10), KW(10), KX(10)
      DIMENSION KY(10), KZ(10), LA(10), LB(10), LC(10), LD(10), LE(10), LF(10), LG(10), LH(10)
      DIMENSION LI(10), LJ(10), LK(10), LL(10), LM(10), LN(10), LO(10), LP(10), LQ(10), LR(10)
      DIMENSION LS(10), LT(10), LU(10), LV(10), LW(10), LX(10), LY(10), LZ(10), MA(10), MB(10)
      DIMENSION MC(10), MD(10), ME(10), MF(10), MG(10), MH(10), MI(10), MJ(10), MK(10), ML(10)
      DIMENSION MM(10), MN(10), MO(10), MP(10), MQ(10), MR(10), MS(10), MT(10), MU(10), MV(10)
      DIMENSION MW(10), MX(10), MY(10), MZ(10), NA(10), NB(10), NC(10), ND(10), NE(10), NF(10)
      DIMENSION NG(10), NH(10), NI(10), NJ(10), NK(10), NL(10), NM(10), NN(10), NO(10), NP(10)
      DIMENSION NQ(10), NR(10), NS(10), NT(10), NU(10), NV(10), NW(10), NX(10), NY(10), NZ(10)
      DIMENSION OA(10), OB(10), OC(10), OD(10), OE(10), OF(10), OG(10), OH(10), OI(10), OJ(10)
      DIMENSION OK(10), OL(10), OM(10), ON(10), OO(10), OP(10), OQ(10), OR(10), OS(10), OT(10)
      DIMENSION OU(10), OV(10), OW(10), OX(10), OY(10), OZ(10), PA(10), PB(10), PC(10), PD(10)
      DIMENSION PE(10), PF(10), PG(10), PH(10), PI(10), PJ(10), PK(10), PL(10), PM(10), PN(10)
      DIMENSION PO(10), PP(10), PQ(10), PR(10), PS(10), PT(10), PU(10), PV(10), PW(10), PX(10)
      DIMENSION PY(10), PZ(10), QA(10), QB(10), QC(10), QD(10), QE(10), QF(10), QG(10), QH(10)
      DIMENSION QI(10), QJ(10), QK(10), QL(10), QM(10), QN(10), QO(10), QP(10), QQ(10), QR(10)
      DIMENSION QS(10), QT(10), QU(10), QV(10), QW(10), QX(10), QY(10), QZ(10), RA(10), RB(10)
      DIMENSION RC(10), RD(10), RE(10), RF(10), RG(10), RH(10), RI(10), RJ(10), RK(10), RL(10)
      DIMENSION RM(10), RN(10), RO(10), RP(10), RQ(10), RR(10), RS(10), RT(10), RU(10), RV(10)
      DIMENSION RW(10), RX(10), RY(10), RZ(10), SA(10), SB(10), SC(10), SD(10), SE(10), SF(10)
      DIMENSION SG(10), SH(10), SI(10), SJ(10), SK(10), SL(10), SM(10), SN(10), SO(10), SP(10)
      DIMENSION SQ(10), SR(10), SS(10), ST(10), SU(10), SV(10), SW(10), SX(10), SY(10), SZ(10)
      DIMENSION TA(10), TB(10), TC(10), TD(10), TE(10), TF(10), TG(10), TH(10), TI(10), TJ(10)
      DIMENSION TK(10), TL(10), TM(10), TN(10), TO(10), TP(10), TQ(10), TR(10), TS(10), TT(10)
      DIMENSION TU(10), TV(10), TW(10), TX(10), TY(10), TZ(10), UA(10), UB(10), UC(10), UD(10)
      DIMENSION UE(10), UF(10), UG(10), UH(10), UI(10), UJ(10), UK(10), UL(10), UM(10), UN(10)
      DIMENSION UO(10), UP(10), UQ(10), UR(10), US(10), UT(10), UY(10), UZ(10), VA(10), VB(10)
      DIMENSION VC(10), VD(10), VE(10), VF(10), VG(10), VH(10), VI(10), VJ(10), VK(10), VL(10)
      DIMENSION VM(10), VN(10), VO(10), VP(10), VQ(10), VR(10), VS(10), VT(10), VU(10), VV(10)
      DIMENSION VW(10), VX(10), VY(10), VZ(10), WA(10), WB(10), WC(10), WD(10), WE(10), WF(10)
      DIMENSION WG(10), WH(10), WI(10), WJ(10), WK(10), WL(10), WM(10), WN(10), WO(10), WP(10)
      DIMENSION WQ(10), WR(10), WS(10), WT(10), WU(10), WV(10), WW(10), WX(10), WY(10), WZ(10)
      DIMENSION XA(10), XB(10), XC(10), XD(10), XE(10), XF(10), XG(10), XH(10), XI(10), XJ(10)
      DIMENSION XK(10), XL(10), XM(10), XN(10), XO(10), XP(10), XQ(10), XR(10), XS(10), XT(10)
      DIMENSION XU(10), XV(10), XW(10), XX(10), XY(10), XZ(10), YA(10), YB(10), YC(10), YD(10)
      DIMENSION YE(10), YF(10), YG(10), YH(10), YI(10), YJ(10), YK(10), YL(10), YM(10), YN(10)
      DIMENSION YO(10), YP(10), YQ(10), YR(10), YS(10), YT(10), YU(10), YV(10), YW(10), YX(10)
      DIMENSION YY(10), YZ(10), ZA(10), ZB(10), ZC(10), ZD(10), ZE(10), ZF(10), ZG(10), ZH(10)
      DIMENSION ZI(10), ZJ(10), ZK(10), ZL(10), ZM(10), ZN(10), ZO(10), ZP(10), ZQ(10), ZR(10)
      DIMENSION ZS(10), ZT(10), ZU(10), ZV(10), ZW(10), ZX(10), ZY(10), ZZ(10)
      END

```

PRECEDING PAGE BLANK-NOT FILMED

```

1 PROGRAM MAIN(INPUT,OUTPUT)
COMMON/RL/CA,HA
EXTERNAL SUHFX
DIMENSION XDAT(5),XDT(5,2)
INTEGER I23K
DATA I220/1/, X221/1./
LASHF(X,Y)=5.*E101(C)*X*Y
C=1.
R=1.
0219=1.0*X221
I23K=1*I220
L=1
M=1
LOC=1
Y=1.0
D=2.0
DO 100 I=1,5
DO 200 J=1,2
XDT(I,J)=I*J
200 CONTINUE
100 CONTINUE
IF(A.EQ.R) X = CA1
X = M*CA
R = F101(I.)
CALL SUR103(3,R,C,Y*1)
CALL SUR103(A,R,C,Y*1)
CALL SUR105(SUHFX,3)
CALL SUR106(SUHFX,3)
CALL SUR208(A,CA)
DO 10 I = 1, 10
K = LOC + 1
10 CONTINUE
K = I + 6
CALL SUR(1.,SUHFX)
I = W201(CA)
CALL SUR215(XDAT,R,C)
CA = 1.
CA = 2.
RA=CA
IF (0219.F0.0) CA=3.*RA
Y22R(I)=6.*R
X22Q=6.
X230 = 1.
IF (CA.F0.L) X230=3.
XDAT(5)=1.*X229+X230
I = W201(I)*X
I = W201(CA)*CA
I = LASHF(2.,5,1)*C
I = F5IM(CA)
CALL SURSTM(X,A,B,D)
STOP
END

```

SYMBOLIC REFERENCE MAP (R=2)

| ENTRY POINTS | DEF LINE | REFERENCES | VARIABLES | SN | TYPE  | RELOCATION | REFS | DEFINITION | REFS    | DEFINITION | REFS | DEFINITION |
|--------------|----------|------------|-----------|----|-------|------------|------|------------|---------|------------|------|------------|
| 44 MAIN      |          |            |           |    |       |            |      |            |         |            |      |            |
|              | 257 A    |            | REAL      |    |       |            | REFS | 23         | 27      | 30         | 51   | 30         |
|              | 246 R    |            | REAL      |    |       |            | REFS | 23         | 26      | 27         | 37   | 27         |
|              | 1 RA     |            | REAL      |    | R1    |            | REFS | 2          | 41      | DEFINED    | 40   | 41         |
|              | 245 C    |            | REAL      |    |       |            | REFS | 26         | 27      | 37         | 2*49 | 2*49       |
|              | 0 CA     |            | REAL      |    | RLK1  |            | REFS | 3          | 24      | 30         | 36   | 30         |
|              | 0 CA1    |            | REAL      |    | R1    |            | REFS | 50         | 38      | 39         | 41   | 39         |
|              | 254 D    |            | REAL      |    |       |            | REFS | 51         | 23      | 17         |      |            |
|              | 1 D21A   |            | REAL      |    | RLK1  |            | REFS | 3          | DEFINED | 11         |      |            |
|              | 247 D219 |            | REAL      |    |       |            | REFS | 41         | DEFINED | 11         |      |            |
|              | 262 I    |            | INTEGER   |    |       |            | REFS | 34         | DEFINED | 31         | 36   | 47         |
|              | 255 I1   |            | INTEGER   |    |       |            | REFS | 50         |         |            |      | 49         |
|              | 233 I220 |            | INTEGER   |    |       |            | REFS | 2*20       | DEFINED | 18         |      |            |
|              | 244 I23A |            | INTEGER   |    |       |            | REFS | 12         | DEFINED | 7          |      |            |
|              | 254 J1   |            | INTEGER   |    |       |            | REFS | 6          | DEFINED | 12         |      |            |
|              | 263 K    |            | INTEGER   |    |       |            | REFS | 2*20       | DEFINED | 19         |      |            |
|              | 250 L    |            | INTEGER   |    |       |            | REFS | 32         | 34      |            |      |            |
|              | 251 LOC  |            | INTEGER   |    |       |            | REFS | 45         | DEFINED | 13         |      |            |
|              | 252 M    |            | INTEGER   |    |       |            | REFS | 32         | DEFINED | 14         |      |            |
|              | 261 R    |            | REAL      |    |       |            | REFS | 24         | DEFINED | 15         |      |            |
|              | 260 X    |            | REAL      |    |       |            | REFS | 42         | DEFINED | 25         |      |            |
|              | 266 XDAT |            | REAL      |    | ARRAY |            | REFS | 2*47       | 51      | DEFINED    | 23   | 24         |
|              | 273 XDT  |            | REAL      |    | ARRAY |            | REFS | 5          | 37      | DEFINED    | 46   |            |
|              | 234 X221 |            | REAL      |    |       |            | REFS | 5          | DEFINED | 20         |      |            |
|              | 264 X229 |            | REAL      |    |       |            | REFS | 11         | DEFINED | 7          |      |            |
|              | 265 X230 |            | REAL      |    |       |            | REFS | 46         | DEFINED | 43         |      |            |
|              | 253 Y    |            | REAL      |    |       |            | REFS | 46         | DEFINED | 44         |      |            |
|              | 2 Y22R   |            | REAL      |    | ARRAY | RLK1       | REFS | 26         | DEFINED | 27         | 45   | 16         |
|              |          |            |           |    |       |            | REFS | 3          | DEFINED | 42         |      |            |

FILE NAMES

| FILE NAMES | P-ODE | TYPE | ARGS | REFERENCES |
|------------|-------|------|------|------------|
| 0 INPUT    |       |      |      |            |
| 20 OUTPUT  |       |      |      |            |
| EXTERNALS  |       |      |      |            |
| E101       | MFAL  | 1    | 25   | 49         |
| FSIM       | MFAL  | 1    | 50   |            |
| SUM        |       | 2    | 35   |            |
| SUMFX      |       | 0    | 4    | 28         |
| SUMSTM     |       | 4    | 51   |            |
| SUM103     |       | 3    | 26   | 27         |
| SUM105     |       | 2    | 28   |            |
| SUM106     |       | 2    | 29   |            |
| SUM208     |       | 3    | 30   |            |
| SUM215     |       | 3    | 37   |            |
| V201       | MFAL  | 1    | 36   | 47         |

INLINE FUNCTIONS TYPE ARGS DEF LINE REFERENCES  
 LARF INTEGER 2 SF 4 49

STATEMENT LABELS

| DEF LINE | REFERENCES |
|----------|------------|
| 0 10     | 31         |
| 0 100    | 1A         |
| 0 200    | 19         |

LOOPS LABEL INDEX FROM-TO LENGTH PROPERTIES  
 54 100 11 1A 22 6R  
 56 200 11 19 21 3R INSTACK  
 114 10 1 31 33 1R INSTACK NOT INNER

COMMON BLOCKS LENGTH  
 R1 2  
 HLK1 R

STATISTICS  
 PROGRAM LENGTH 165  
 BUFFER LENGTH 40R 32  
 COMMON LENGTH 12M 10  
 COMMON SC= USED 55000R



\*\*\* 04/24/78 SCOPF 2.1.4 P R L VFR 004 \*\*\* 05/26/78 7H166

SYS DEVICFS P19/ 4/PE FLS=377K FLL=1750K MXS=240K MXL=1300K MXH=1300H

HH.MM.SS CPU SECOND ORIGIN

```
0R.40.11 00000.002 MF7. AF RRL NOS/HE 1.2 L447 VERSION 4.1 05/15/78
0R.40.11 00000.003 JOR. -SRL4H,STMFZ,T40. EXECUTE DAVE PROC
0R.40.12 00000.025 JOR. -ACCOINT(LD***))
0R.40.12 00000.028 MF7. -ATTACH(OLNPL,PLTFSTFLC,LD=SRLJL)
0R.40.12 00000.029 LOD. PF254 - CYCLE 2 ATTACHED FROM SN=SYSTEM
0R.40.14 00000.077 USR. -UPDATE(F,C=TEST,D,R)
0R.40.14 00000.078 LOD. UPDATE COMPLETED
0R.40.14 00000.091 MF7. -RFGIN,DAVF,DAVE,INPUT,TEST.
0R.40.14 00000.097 MF7. PF446 - PFMACRO - ATTACH - DAVE - DAVE
0R.40.15 00000.119 JOR. PF254 - CYCLE 1 ATTACHED FROM SN=SYSTEM
0R.40.15 00000.120 LOD. -MAP(OFF)
0R.40.16 00000.180 USR. -FTN(I=Z7CCLAA,R=COMP,L=0)
0R.40.16 00000.185 JOR. .054 CP SECONDS COMPILATION TIME
0R.40.16 00000.185 MF7. -ATTACH(PHOR,LD=SRLJL)
0R.40.16 00000.189 MF7. PF053 - LFN IS PH08
0R.40.16 00000.192 LOD. PF254 - CYCLE 2 ATTACHED FROM SN=SYSTEM
0R.40.17 00000.334 MF7. -LOAD(COMP)
0R.40.17 00000.335 MF7. -PH08(INPUT,TFST)
0R.40.17 00000.335 USR. LD610 - FLS REQUIRED TO LOAD - 0012275 OU.COG
0R.40.17 00000.459 USR. LD603 - EXECUTION INITIATED OS.EXP
0R.40.17 00000.459 USR. FORTRAN LIBRARY 452 08/04/77
0R.40.17 00000.464 JOR. STOP
0R.40.17 00000.464 MF7. .123 CP SECONDS EXECUTION TIME
0R.40.18 00000.468 MF7. -ATTACH(PH1R,LD=SRLJL)
0R.40.18 00000.469 JOR. PF053 - LFN IS PH18
0R.40.18 00000.473 MF7. PF254 - CYCLE 2 ATTACHED FROM SN=SYSTEM
0R.40.18 00000.474 JOR. -ATTACH(COMDAT,LD=SRLJL)
0R.40.18 00000.474 MF7. PF053 - LFN IS COMDAT
0R.40.18 00000.478 MF7. PF254 - CYCLE 3 ATTACHED FROM SN=SYSTEM
0R.40.18 00000.482 LOD. -ATTACH(DBLIR,LD=SRLJL)
0R.40.18 00001.031 MF7. PF053 - LFN IS DBLIR
0R.40.18 00001.032 MF7. PF254 - CYCLE 1 ATTACHED FROM SN=SYSTEM
0R.40.18 00001.032 USR. -LIBRARY(DBLIR)
0R.40.19 00001.483 USR. -PH1R.
0R.40.19 00001.483 USR. LD610 - FLS REQUIRED TO LOAD - 0020655 OU.COG
0R.40.19 00001.483 USR. LD603 - EXECUTION INITIATED OS.EXP
0R.40.19 00002.483 USR. FORTRAN LIBRARY 452 08/04/77
0R.40.24 00002.483 USR. STOP
0R.40.24 00002.483 USR. 1.449 CP SECONDS EXECUTION TIME
0R.40.24 00002.495 JOR. -RETURN(PH1H)
0R.40.24 00002.496 MF7. -ATTACH(PH2H,LD=SRLJL)
0R.40.24 00002.499 MF7. PF053 - LFN IS PH2H
0R.40.24 00002.500 LOD. PF254 - CYCLE 1 ATTACHED FROM SN=SYSTEM
0R.40.26 00003.012 MF7. -PH2H.
0R.40.26 00003.015 MF7. LD610 - FLS REQUIRED TO LOAD - 0021562 OU.COG
0R.40.26 00003.015 USR. LD603 - EXECUTION INITIATED OS.EXP
0R.40.26 00005.575 USR. FORTRAN LIBRARY 452 08/04/77
0R.40.26 00005.575 USR. STOP
0R.40.26 00005.576 LOD. 2.557 CP SECONDS EXECUTION TIME
0R.40.26 00005.588 JOR. -RETURN(PH2R)
0R.40.26 00005.588 MF7. -ATTACH(PH3R,LD=SRLJL)
0R.40.26 00005.592 MF7. PF053 - LFN IS PH3H
0R.40.26 00005.592 LOD. PF254 - CYCLE 1 ATTACHED FROM SN=SYSTEM
0R.40.26 00005.592 USR. -PH3R.
0R.40.26 00005.875 MF7. LD610 - FLS REQUIRED TO LOAD - 0015064 OU.COG
0R.40.26 00005.877 MF7. LD603 - EXECUTION INITIATED OS.EXP
0R.40.26 00005.877 USR. FORTRAN LIBRARY 452 08/04/77
```

08.40.40 00006.H15 USR.

STOP

```
08.40.40 00006.H15 USR.          .435 CP SECONDS EXECUTION TIME
08.40.40 00006.H15 LOD.          -RETURN(PH3M)
08.40.40 00006.H23 LOD.          -REVFPT.
08.40.41 00006.H39 MF7.          RM770 - MAXIMUM ACTIVE FILES          11
08.40.41 00006.H39 MF7.          RM771 - OPEN/CLOSE CALLS            129
08.40.41 00006.H39 MF7.          RM772 - DATA TRANSFER CALLS       3,988
08.40.41 00006.H39 MF7.          RM773 - CONTROL/POSITIONING CALLS  131
08.40.41 00006.H39 MF7.          RM774 - RM DATA TRANSFER CALLS    2,321
08.40.41 00006.H39 MF7.          RM775 - RM CONTROL/POSITIONING CALLS 273
08.40.41 00006.H39 MF7.          RM776 - QUEUE MANAGER CALLS       448
08.40.41 00006.H40 MF7.          RM777 - RECALL CALLS              356
08.40.41 00006.H40 MF7.          SCM                               201,028 KWS
08.40.41 00006.H40 MF7.          LCM                               174,415 KWS
08.40.41 00006.H40 MF7.          I/O                               0,414 MW
08.40.41 00006.H40 MF7.          RMS                               0,782 MWS
08.40.41 00006.H41 MF7.          USER                             4,538 SEC
08.40.41 00006.H41 MF7.          JOB                               6,443 SEC
08.40.41 00006.H41 MF7.          DIO                               1 030,774 KW
08.40.41 00006.H41 MF7.          SC050 - 000001 SC/LC SWAPS
```

\*\*\*\*\* SHL4824 0002039 LINES PRINTED

.....  
\*  
\* DAVE TERMINATION NORMAL \*  
\*  
\*.....

.....  
\* NOTE -- FOR MISSING SUBPROGRAMS THE FOLLOWING I/O BEHAVIOR \*  
\* WAS BEEN SIMULATED. \*  
\*.....

- A. FOR FUNCTION SUBPROGRAMS, THE FUNCTION NAME HAS BEEN CLASSIFIED AS STRICT OUTPUT AND ALL ARGUMENTS AS STRICT INPUT, NON-OUTPUT.
- B. FOR SUBROUTINE SUBPROGRAMS, ALL ARGUMENTS HAVE BEEN CLASSIFIED AS STRICT INPUT, NON-OUTPUT.

.....  
\* A SIMULATED SUBPROGRAM IS ASSUMED TO USE NO COMMON \*  
\* VARIABLES. THE NUMBER AND DIMENSIONS OF ITS DUMMY \*  
\* ARGUMENTS HAVE BEEN INFERRED FROM THE FIRST INVOCATION \*  
\* OF THE SUBPROGRAM BY THE PROGRAM UNIT INDICATED BELOW. \*  
\*.....

| SIMULATED SUBPROGRAM | CALLER      |
|----------------------|-------------|
| -----                | -----       |
| --*FSIMP--           | --*SYSMAIN* |
| --*SIHSH*            | --*SYSMAIN* |

USER OPTIONS SPECIFIED THIS RUN  
-----

1. SIMULATE I/O BEHAVIOR FOR MISSING SUBPROGRAMS (SI=ON).
2. RE-START OF PREVIOUS RUN (RF=OFF).
3. SUPPRESS DIAGNOSTICS (SU=OFF).

DIAGNOSTIC SUMMARY -- PART 1

| SUBPROGRAM | FREQUENCY |          |          |
|------------|-----------|----------|----------|
|            | ERRORS    | WARNINGS | MESSAGES |
| SYSMAIN    | 14        | 42       | 5        |
| HLKDATA    |           |          | 1        |
| E101       | 1         | 4        | 1        |
| SUR103     |           | 2        | 2        |
| SUR302     | 1         | 6        | 2        |
| SIM105     |           |          | 1        |
| SIM106     | 1         | 1        | 2        |
| SUR200     |           |          | 1        |
| W201       |           | 3        | 1        |
| SUR215     |           | 1        | 1        |
| SUR        |           | 4        | 1        |
| FIN        |           | 1        | 1        |
| FSIM       |           |          | 1        |
| SURSJM     |           |          | 1        |

DIAGNOSTIC SUMMARY -- PART 2

| ERRORS    |           | WARNINGS  |           | MESSAGES  |           |
|-----------|-----------|-----------|-----------|-----------|-----------|
| IDENT.NO. | FREQUENCY | IDENT.NO. | FREQUENCY | IDENT.NO. | FREQUENCY |
| 101       | 1         | 201       | 1         | 301       | 2         |
| 103       | 5         | 202       | 1         | 302       | 3         |
| 105       | 1         | 203       | 2         | 303       | 2         |
| 106       | 2         | 204       | 5         | 304       | 14        |
| 108       | 2         | 205       | 1         |           |           |
| 109       | 2         | 206       | 1         |           |           |
| 110       | 4         | 208       | 1         |           |           |
| 111       | 2         | 209       | 1         |           |           |
| 112       | 2         | 210       | 3         |           |           |
|           |           | 211       | 1         |           |           |
|           |           | 213       | 2         |           |           |
|           |           | 214       | 3         |           |           |
|           |           | 215       | 1         |           |           |
|           |           | 216       | 5         |           |           |
|           |           | 217       | 2         |           |           |
|           |           | 218       | 1         |           |           |
|           |           | 219       | 1         |           |           |
|           |           | 220       | 1         |           |           |
|           |           | 221       | 1         |           |           |
|           |           | 222       | 1         |           |           |
|           |           | 223       | 2         |           |           |
|           |           | 224       | 1         |           |           |
|           |           | 225       | 1         |           |           |
|           |           | 226       | 1         |           |           |
|           |           | 227       | 1         |           |           |
|           |           | 228       | 1         |           |           |
|           |           | 229       | 8         |           |           |
|           |           | 230       | 3         |           |           |
|           |           | 231       | 2         |           |           |
|           |           | 232       | 2         |           |           |
|           |           | 233       | 2         |           |           |
|           |           | 234       | 1         |           |           |
|           |           | 235       | 1         |           |           |
|           |           | 237       | 3         |           |           |

CALL GRAPH  
-----

| SUMPROGRAM<br>----- | CALLED BY<br>-----          | CALLS<br>-----  |
|---------------------|-----------------------------|---|
| SYSMAIN             |                             | E101<br>SUB103<br>SUB105<br>SUB106<br>SUB208<br>W201<br>SUB215<br>SUB<br>FSIM<br>SUBSIM |
| -----               | -----                       | -----   |
| E101                | SYSMAIN<br>SUB106           |   |
| -----               | -----                       | -----   |
| SUB103              | SYSMAIN                     | SUB302  |
| -----               | -----                       | -----   |
| SUB302              | SUB103                      | SUB106  |
| -----               | -----                       | -----   |
| SUB105              | SYSMAIN                     | SUB106  |
| -----               | -----                       | -----   |
| SUB106              | SYSMAIN<br>SUB302<br>SUB105 | E101  |
| -----               | -----                       | -----   |
| SUB208              | SYSMAIN                     |   |
| -----               | -----                       | -----   |
| W201                | SYSMAIN                     |   |
| -----               | -----                       | -----   |
| SUB215              | SYSMAIN                     |   |
| -----               | -----                       | -----   |
| SUB                 | SYSMAIN                     |   |
| -----               | -----                       | -----   |
| FUN                 |                             |   |
| -----               | -----                       | -----   |
| FSIM                | SYSMAIN                     |   |
| -----               | -----                       | -----   |
| SUBSIM              | SYSMAIN                     |   |
| -----               | -----                       | -----   |

SOURCE PROGRAM LISTING

\$ IN THE CONTINUATION FIELD INDICATES THE EXPANSION OF THE LOGICAL IF STATEMENT ON THE PREVIOUS LINE

| BLOCK | SOURCE                      |
|-------|-----------------------------|
| 1     | PROGRAM MAIN(INPUT,OUTPUT)  |
| 1     | COMMON/H1/CA1,RA            |
| 1     | COMMON/BLK1/CA,D219,Y228(A) |
| 1     | EXTERNAL SUBEX              |
| 1     | DIMENSION XDAT(5),XDT(5,2)  |
| 1     | INTEGER IP36                |
| 1     | DATA I220(1), X221(1),      |
| 1     | IASRF(X,Y)=5.*E101(C)*X*Y   |
| 2     | C=1.                        |
| 3     | H=1.                        |
| 4     | D219=1.0+X221               |
| 5     | IP36=1+I220                 |
| 6     | L=1                         |
| 7     | LOC=1                       |
| 8     | M=1                         |
| 9     | Y=1.0                       |
| 10    | N=2.0                       |
| 11    | DO 100 I1=1,5               |
| 12    | DO 200 J1=1,2               |
| 13    | XDT(I1,J1)=I1*J1            |
| 14    | 200 CONTINUE                |
| 15    | 100 CONTINUE                |
| 16    | IF(A.EQ.H)                  |
| 17    | \$ X = CA1                  |
| 18    | X = M*CA                    |
| 19    | R = F101(1.)                |
| 20    | CALL SUB103(3,R+C,Y+1)      |
| 21    | CALL SUB103(A,R+C,Y+1)      |
| 22    | CALL SUB105(SUBFX,3)        |
| 23    | CALL SUB106(SUBFX,3)        |
| 24    | CALL SUB20R(A,CA)           |
| 25    | DO 10 I = 1, 10             |
| 26    | K = LOC + 1                 |
| 27    | 10 CONTINUE                 |
| 28    | K = I + 6                   |
| 29    | CALL SUB(1.,SUBFX)          |
| 30    | I = W201(CA)                |
| 31    | CALL SUB215(XDAT,R,C)       |
| 32    | CA = 1.                     |
| 33    | CA = 2.                     |
| 34    | RA=CA                       |
| 35    | IF(D219.EQ.0)               |
| 36    | \$ CA=3.+HA                 |
| 37    | Y228(1)=6.*R                |
| 38    | X229=6.                     |
| 39    | X230 = 1.                   |
| 40    | IF(CA.EQ.L)                 |
| 41    | \$ X230=3.                  |
| 42    | XDAT(5)=1.*X229*X230        |
| 43    | I = W201(X)+X               |
| 44    | I = W201(CA)+CA             |
| 45    | I = IASRF(2.,5.)*C          |
| 46    | I = FSIM(CA)                |
| 47    | CALL SUBSIM(X,A,H,D)        |
| 48    | STOP                        |
| 1     | END                         |

F H W O R S  
- - - - -

| ERROR<br>NUMBER<br>----- | DESCRIPTION<br>-----   |
|--------------------------|--|
| ** 103 **                | <p>BLOCK NO. 19<br/>AN ACTUAL ARGUMENT IS AN EXPRESSION OR CONSTANT, YET THE<br/>CORRESPONDING DUMMY ARGUMENT IS ASSIGNED A VALUE ON ALL PATHS.<br/>CALLING SUBPROGRAM CALLED SUBPROGRAM</p> <p style="text-align: center;">--*SYSMAIN*--      ----*E101*--<br/>REAL                      ----*A*-----</p> <p style="text-align: center;">ARGUMENT                      REAL                      1<br/>POSITION                      1                      1</p>               |
| ** 103 **                | <p>BLOCK NO. 20<br/>AN ACTUAL ARGUMENT IS AN EXPRESSION OR CONSTANT, YET THE<br/>CORRESPONDING DUMMY ARGUMENT IS ASSIGNED A VALUE ON ALL PATHS.<br/>CALLING SUBPROGRAM CALLED SUBPROGRAM</p> <p style="text-align: center;">--*SYSMAIN*--      ----*SUI103*--<br/>INTEGER                      ----*I*-----</p> <p style="text-align: center;">ARGUMENT                      INTEGER                      1<br/>POSITION                      1                      1</p>       |
| ** 103 **                | <p>BLOCK NO. 20<br/>AN ACTUAL ARGUMENT IS AN EXPRESSION OR CONSTANT, YET THE<br/>CORRESPONDING DUMMY ARGUMENT IS ASSIGNED A VALUE ON ALL PATHS.<br/>CALLING SUBPROGRAM CALLED SUBPROGRAM</p> <p style="text-align: center;">--*SYSMAIN*--      ----*SUI103*--<br/>EXPRESSION                      ----*X*-----</p> <p style="text-align: center;">ARGUMENT                      EXPRESSION                      2<br/>POSITION                      2                      2</p> |
| ** 103 **                | <p>BLOCK NO. 21<br/>AN ACTUAL ARGUMENT IS AN EXPRESSION OR CONSTANT, YET THE<br/>CORRESPONDING DUMMY ARGUMENT IS ASSIGNED A VALUE ON ALL PATHS.<br/>CALLING SUBPROGRAM CALLED SUBPROGRAM</p> <p style="text-align: center;">--*SYSMAIN*--      ----*SUI103*--<br/>EXPRESSION                      ----*X*-----</p> <p style="text-align: center;">ARGUMENT                      EXPRESSION                      2<br/>POSITION                      2                      2</p> |
| ** 103 **                | <p>BLOCK NO. 23<br/>AN ACTUAL ARGUMENT IS AN EXPRESSION OR CONSTANT, YET THE<br/>CORRESPONDING DUMMY ARGUMENT IS ASSIGNED A VALUE ON ALL PATHS.<br/>CALLING SUBPROGRAM CALLED SUBPROGRAM</p> <p style="text-align: center;">--*SYSMAIN*--      ----*SUI106*--<br/>INTEGER                      ----*Z*-----</p> <p style="text-align: center;">ARGUMENT                      INTEGER                      2<br/>POSITION                      2                      2</p>       |

**\*\* 105 \*\*** BLOCK NO. 22  
 AN ACTUAL ARGUMENT IS A PROCEDURE DECLARED EXTERNAL, YET THE  
 CORRESPONDING DUMMY ARGUMENT IS REFERENCED AS A VARIABLE  
 ON ALL PATHS.

|          |                    |            |                   |      |
|----------|--------------------|------------|-------------------|------|
|          | CALLING SUBPROGRAM |            | CALLED SUBPROGRAM |      |
|          | --*SYSMAIN*        | --*SURI05* |                   |      |
| ARGUMENT | --*SUREX*          | ----       | ----              | ---- |
| POSITION | 1                  | 1          |                   |      |

**\*\* 106 \*\*** BLOCK NO. 22  
 AN ACTUAL ARGUMENT IS A PROCEDURE DECLARED EXTERNAL, YET THE  
 CORRESPONDING DUMMY ARGUMENT, USED AS A VARIABLE, IS ASSIGNED  
 A VALUE ON ALL PATHS.

|          |                    |            |                   |      |
|----------|--------------------|------------|-------------------|------|
|          | CALLING SUBPROGRAM |            | CALLED SUBPROGRAM |      |
|          | --*SYSMAIN*        | --*SURI05* |                   |      |
| ARGUMENT | --*SUREX*          | ----       | ----              | ---- |
| POSITION | 1                  | 1          |                   |      |

**\*\* 106 \*\*** BLOCK NO. 23  
 AN ACTUAL ARGUMENT IS A PROCEDURE DECLARED EXTERNAL, YET THE  
 CORRESPONDING DUMMY ARGUMENT, USED AS A VARIABLE, IS ASSIGNED  
 A VALUE ON ALL PATHS.

|          |                    |            |                   |      |
|----------|--------------------|------------|-------------------|------|
|          | CALLING SUBPROGRAM |            | CALLED SUBPROGRAM |      |
|          | --*SYSMAIN*        | --*SURI06* |                   |      |
| ARGUMENT | --*SUREX*          | ----       | ----              | ---- |
| POSITION | 1                  | 1          |                   |      |

**\*\* 10A \*\*** BLOCK NO. 30  
 A SUBPROGRAM REFERENCE CAUSES DUMMY ARGUMENT ----\*X\*----  
 TO BECOME ASSOCIATED WITH A COMMON VARIABLE IN THE CALLED  
 SUBPROGRAM. ----\*X\*---- IS ASSIGNED A VALUE ON ALL PATHS.

|                 |                    |      |                   |      |
|-----------------|--------------------|------|-------------------|------|
|                 | CALLING SUBPROGRAM |      | CALLED SUBPROGRAM |      |
|                 | --*SYSMAIN*        | ---- | --*W201*          |      |
| ARGUMENT        | ----               | ---- | ----              | ---- |
| COMMON VARIABLE | ----               | ---- | ----              | ---- |

**\*\* 10A \*\*** BLOCK NO. 44  
 A SUBPROGRAM REFERENCE CAUSES DUMMY ARGUMENT ----\*X\*----  
 TO BECOME ASSOCIATED WITH A COMMON VARIABLE IN THE CALLED  
 SUBPROGRAM. ----\*X\*---- IS ASSIGNED A VALUE ON ALL PATHS.

|                 |                    |      |                   |      |
|-----------------|--------------------|------|-------------------|------|
|                 | CALLING SUBPROGRAM |      | CALLED SUBPROGRAM |      |
|                 | --*SYSMAIN*        | ---- | --*W201*          |      |
| ARGUMENT        | ----               | ---- | ----              | ---- |
| COMMON VARIABLE | ----               | ---- | ----              | ---- |

**\*\* 109 \*\*** COMMON VARIABLE ----\*Y22R\*-- IN COMMON BLOCK ----\*HLK1\*-- IS  
 REFERENCED ON ALL PATHS IN THE MAIN PROGRAM, YET IT HAS NOT  
 PREVIOUSLY BEEN ASSIGNED A VALUE, NOR HAS IT BEEN INITIALIZED  
 IN BLOCK DATA. (SEE NOTE 1)

**\*\* 109 \*\*** COMMON VARIABLE ----\*CA\*-- IN COMMON BLOCK ----\*HLK1\*-- IS  
 REFERENCED ON ALL PATHS IN THE MAIN PROGRAM, YET IT HAS NOT  
 PREVIOUSLY BEEN ASSIGNED A VALUE, NOR HAS IT BEEN INITIALIZED  
 IN BLOCK DATA. (SEE NOTE 1)

- \*\* 110 \*\* COMMON VARIABLE -----M\*----- IS REFERENCED ON ALL PATHS IN  
CALLED SUBPROGRAM ---\*E101\*---, YET IS NOT INITIALIZED. IT  
DOES NOT APPEAR IN BLOCK DATA, AND ITS COMMON BLOCK ---\*F110\*---  
IS NOT AVAILABLE TO CALLING SUBPROGRAM --\*SYSMAIN\*-- (SEE  
NOTE 1)**
- \*\* 110 \*\* COMMON VARIABLE -----B\*----- IS REFERENCED ON ALL PATHS IN  
CALLED SUBPROGRAM --\*SUB103\*--, YET IS NOT INITIALIZED. IT  
DOES NOT APPEAR IN BLOCK DATA, AND ITS COMMON BLOCK ---\*HLK\*---  
IS NOT AVAILABLE TO CALLING SUBPROGRAM --\*SYSMAIN\*-- (SEE  
NOTE 1)**
- \*\* 110 \*\* COMMON VARIABLE -----D\*----- IS REFERENCED ON ALL PATHS IN  
CALLED SUBPROGRAM --\*SUM200\*--, YET IS NOT INITIALIZED. IT  
DOES NOT APPEAR IN BLOCK DATA, AND ITS COMMON BLOCK ---\*BLK\*---  
IS NOT AVAILABLE TO CALLING SUBPROGRAM --\*SYSMAIN\*-- (SEE  
NOTE 1)**
- \*\* 111 \*\* CONTROL VARIABLE -----I\*----- BECOMES UNDEFINED UPON SATISFACTION  
OF ITS DO LOOP AT BLOCK NO. 27, YET IS REFERENCED ON ALL  
PATHS THEREAFTER.  
ONE SUCH PATH, INDICATED BY BLOCK NUMBERS, IS  
27 28**
- \*\* 112 \*\* LOCAL VARIABLE ---\*XDAT\*--- IS REFERENCED BEFORE BEING ASSIGNED  
A VALUE ON ALL PATHS.  
ONE SUCH PATH, INDICATED BY BLOCK NUMBERS, IS  
1 - 31**
- \*\* 112 \*\* LOCAL VARIABLE -----A\*----- IS REFERENCED BEFORE BEING ASSIGNED  
A VALUE ON ALL PATHS.  
ONE SUCH PATH, INDICATED BY BLOCK NUMBERS, IS  
1 - 16**

W A R N I N G S

| WARNING<br>NUMBER<br>----- | DESCRIPTION<br>-----  |
|----------------------------|---|
| <b>** 203 **</b>           | BLOCK NO. 20<br>AN ACTUAL ARGUMENT IS AN EXPRESSION OR CONSTANT, YET THE<br>CORRESPONDING DUMMY ARGUMENT IS ASSIGNED A VALUE ON SOME PATHS.<br>CALLING SUBPROGRAM CALLED SUBPROGRAM<br><div style="display: flex; justify-content: space-around; margin-top: 5px;"> <div style="text-align: center;">             --*SYSMAIN*--<br/>             ARGUMENT<br/>POSITION           </div> <div style="text-align: center;">             EXPRESSION<br/>3           </div> <div style="text-align: center;">             ---*SUM103*--<br/>             -----Y*-----<br/>3           </div> </div> |
| <b>** 203 **</b>           | BLOCK NO. 21<br>AN ACTUAL ARGUMENT IS AN EXPRESSION OR CONSTANT, YET THE  |

CORRESPONDING DUMMY ARGUMENT IS ASSIGNED A VALUE ON SOME PATHS.

|          |                    |                   |
|----------|--------------------|-------------------|
|          | CALLING SUBPROGRAM | CALLED SUBPROGRAM |
|          | --*SYSMAIN*        | --*SUR103*        |
| ARGUMENT | EXPRESSION         | ----              |
| POSITION | 3                  | 3                 |

\*\* 204 \*\* BLOCK NO. 19  
 AN ACTUAL ARGUMENT IS AN EXPRESSION OR CONSTANT, YET THE  
 CORRESPONDING DUMMY ARGUMENT IS NEVER REFERENCED.

|          |                    |                   |
|----------|--------------------|-------------------|
|          | CALLING SUBPROGRAM | CALLED SUBPROGRAM |
|          | --*SYSMAIN*        | --*E101*          |
| ARGUMENT | REAL               | ----              |
| POSITION | 1                  | 1                 |

\*\* 204 \*\* BLOCK NO. 20  
 AN ACTUAL ARGUMENT IS AN EXPRESSION OR CONSTANT, YET THE  
 CORRESPONDING DUMMY ARGUMENT IS NEVER REFERENCED.

|          |                    |                   |
|----------|--------------------|-------------------|
|          | CALLING SUBPROGRAM | CALLED SUBPROGRAM |
|          | --*SYSMAIN*        | --*SUR103*        |
| ARGUMENT | INTEGER            | ----              |
| POSITION | 1                  | 1                 |

\*\* 204 \*\* BLOCK NO. 20  
 AN ACTUAL ARGUMENT IS AN EXPRESSION OR CONSTANT, YET THE  
 CORRESPONDING DUMMY ARGUMENT IS NEVER REFERENCED.

|          |                    |                   |
|----------|--------------------|-------------------|
|          | CALLING SUBPROGRAM | CALLED SUBPROGRAM |
|          | --*SYSMAIN*        | --*SUR103*        |
| ARGUMENT | EXPRESSION         | ----              |
| POSITION | 2                  | 2                 |

\*\* 204 \*\* BLOCK NO. 21  
 AN ACTUAL ARGUMENT IS AN EXPRESSION OR CONSTANT, YET THE  
 CORRESPONDING DUMMY ARGUMENT IS NEVER REFERENCED.

|          |                    |                   |
|----------|--------------------|-------------------|
|          | CALLING SUBPROGRAM | CALLED SUBPROGRAM |
|          | --*SYSMAIN*        | --*SUR103*        |
| ARGUMENT | EXPRESSION         | ----              |
| POSITION | 2                  | 2                 |

\*\* 204 \*\* BLOCK NO. 23  
 AN ACTUAL ARGUMENT IS AN EXPRESSION OR CONSTANT, YET THE  
 CORRESPONDING DUMMY ARGUMENT IS NEVER REFERENCED.

|          |                    |                   |
|----------|--------------------|-------------------|
|          | CALLING SUBPROGRAM | CALLED SUBPROGRAM |
|          | --*SYSMAIN*        | --*SUR106*        |
| ARGUMENT | INTEGER            | ----              |
| POSITION | 2                  | 2                 |

\*\* 205 \*\* BLOCK NO. 29  
 AN ACTUAL ARGUMENT IS A PROCEDURE DECLARED EXTERNAL, YET THE  
 CORRESPONDING DUMMY ARGUMENT IS REFERENCED AS A VARIABLE ON  
 SOME PATHS.

|          |                    |                   |
|----------|--------------------|-------------------|
|          | CALLING SUBPROGRAM | CALLED SUBPROGRAM |
|          | --*SYSMAIN*        | --*SUR*           |
| ARGUMENT | --*SURFX*          | ----              |
| POSITION | 2                  | 2                 |

\*\* 206 \*\* BLOCK NO. 20

AN ACTUAL ARGUMENT IS A PROCEDURE DECLARED EXTERNAL, YET THE

CORRESPONDING DUMMY ARGUMENT, USED AS A VARIABLE, IS ASSIGNED  
A VALUE ON SOME PATHS.

|          |                    |                   |
|----------|--------------------|-------------------|
|          | CALLING SUBPROGRAM | CALLED SUBPROGRAM |
|          | --*SYSMAIN*        | ---*SUB*          |
| ARGUMENT | --*SUBEX*          | ----*B*           |
| POSITION | 2                  | 2                 |

\*\* 208 \*\* BLOCK NO. 24  
A SUBPROGRAM REFERENCE CAUSES DUMMY ARGUMENT ----\*X\*----  
TO BECOME ASSOCIATED WITH A COMMON VARIABLE IN THE CALLED  
SUBPROGRAM. ----\*X\*---- IS ASSIGNED A VALUE ON SOME PATHS.

|                 |                    |                   |
|-----------------|--------------------|-------------------|
|                 | CALLING SUBPROGRAM | CALLED SUBPROGRAM |
|                 | --*SYSMAIN*        | ---*SUB208*       |
| ARGUMENT        | ----*CA*           | ----*X*           |
| COMMON VARIABLE | ----*CA*           | ----*CA*          |

\*\* 209 \*\* COMMON VARIABLE ----\*CA1\*--- IN COMMON BLOCK ----\*R1\*--- IS  
REFERENCED ON SOME PATHS IN THE MAIN PROGRAM, YET IT HAS NOT  
PREVIOUSLY BEEN ASSIGNED A VALUE, NOR HAS IT BEEN INITIALIZED  
IN BLOCK DATA. (SEE NOTE 1)

\*\* 210 \*\* COMMON VARIABLE ----\*C\*---- IS REFERENCED ON SOME PATHS IN  
CALLED SUBPROGRAM --\*SUB103\*--, YET IS NOT INITIALIZED.  
IT DOES NOT APPEAR IN BLOCK DATA, AND ITS COMMON BLOCK  
----\*BLK\*--- IS NOT AVAILABLE TO CALLING SUBPROGRAM  
--\*SYSMAIN\*-. (SEE NOTE 1)

\*\* 210 \*\* COMMON VARIABLE ----\*D\*---- IS REFERENCED ON SOME PATHS IN  
CALLED SUBPROGRAM --\*SUB103\*--, YET IS NOT INITIALIZED.  
IT DOES NOT APPEAR IN BLOCK DATA, AND ITS COMMON BLOCK  
----\*BLK\*--- IS NOT AVAILABLE TO CALLING SUBPROGRAM  
--\*SYSMAIN\*-. (SEE NOTE 1)

\*\* 210 \*\* COMMON VARIABLE ----\*R\*---- IS REFERENCED ON SOME PATHS IN  
CALLED SUBPROGRAM --\*SUB208\*--, YET IS NOT INITIALIZED.  
IT DOES NOT APPEAR IN BLOCK DATA, AND ITS COMMON BLOCK  
----\*BLK\*--- IS NOT AVAILABLE TO CALLING SUBPROGRAM  
--\*SYSMAIN\*-. (SEE NOTE 1)

\*\* 213 \*\* BLOCK NO. 21  
CORRESPONDING ARGUMENTS HAVE DIFFERENT DATA TYPES.

|           |                    |                   |
|-----------|--------------------|-------------------|
|           | CALLING SUBPROGRAM | CALLED SUBPROGRAM |
|           | --*SYSMAIN*        | ---*SUB103*       |
| ARGUMENT  | ----*A*----        | ----*I*----       |
| POSITION  | 1                  | 1                 |
| DATA TYPE | REAL               | INTEGER           |

\*\* 213 \*\* BLOCK NO. 23  
CORRESPONDING ARGUMENTS HAVE DIFFERENT DATA TYPES.

|           |                    |                   |
|-----------|--------------------|-------------------|
|           | CALLING SUBPROGRAM | CALLED SUBPROGRAM |
|           | --*SYSMAIN*        | ---*SUB106*       |
| ARGUMENT  | INTEGER            | ----*Z*----       |
| POSITION  | 2                  | 2                 |
| DATA TYPE | INTEGER            | REAL              |

\*\* 214 \*\* CORRESPONDING COMMON VARIABLES IN COMMON BLOCK ---HLK1---  
HAVE DIFFERENT DATA TYPES.

|           |                    |                   |
|-----------|--------------------|-------------------|
|           | CALLING SUBPROGRAM | CALLED SUBPROGRAM |
|           | --SYSMAIN--        | --SUR103--        |
| VARIABLE  | ----*CA*---        | ----*K*---        |
| DATA TYPE | REAL               | INTEGER           |

\*\* 214 \*\* CORRESPONDING COMMON VARIABLES IN COMMON BLOCK ---HLK1---  
HAVE DIFFERENT DATA TYPES.

|           |                    |                   |
|-----------|--------------------|-------------------|
|           | CALLING SUBPROGRAM | CALLED SUBPROGRAM |
|           | --SYSMAIN--        | --SUR103--        |
| VARIABLE  | ----*D?10*---      | ----*K*---        |
| DATA TYPE | REAL               | INTEGER           |

\*\* 214 \*\* CORRESPONDING COMMON VARIABLES IN COMMON BLOCK ---HLK1---  
HAVE DIFFERENT DATA TYPES.

|           |                    |                   |
|-----------|--------------------|-------------------|
|           | CALLING SUBPROGRAM | CALLED SUBPROGRAM |
|           | --SYSMAIN--        | --SUR103--        |
| VARIABLE  | ----*Y220*---      | ----*K*---        |
| DATA TYPE | REAL               | INTEGER           |

\*\* 215 \*\* BLOCK NO. 31  
CORRESPONDING ARGUMENTS HAVE DIFFERENT DIMENSIONALITY.

|            |                    |                   |
|------------|--------------------|-------------------|
|            | CALLING SUBPROGRAM | CALLED SUBPROGRAM |
|            | --SYSMAIN--        | --SUR215--        |
| ARGUMENT   | ----*XDAT*---      | ----*XDAT*---     |
| POSITION   | 1                  | 1                 |
| DIMENSIONS | 1                  | 2                 |

\*\* 216 \*\* COMMON VARIABLE ----\*M\*---- IS ASSIGNED A VALUE ON ALL PATHS  
IN CALLED SUBPROGRAM ---F101---, YET ITS COMMON BLOCK  
----\*E110\*--- IS NOT AVAILABLE TO CALLING SUBPROGRAM --SYSMAIN--.  
HENCE, A COMPUTED VALUE WILL BE LOST. (SEE NOTE 1)

\*\* 216 \*\* COMMON VARIABLE ----\*R\*---- IS ASSIGNED A VALUE ON ALL PATHS  
IN CALLED SUBPROGRAM --SUR103--, YET ITS COMMON BLOCK  
----\*RLK\*--- IS NOT AVAILABLE TO CALLING SUBPROGRAM --SYSMAIN--.  
HENCE, A COMPUTED VALUE WILL BE LOST. (SEE NOTE 1)

\*\* 216 \*\* COMMON VARIABLE ----\*D\*---- IS ASSIGNED A VALUE ON ALL PATHS  
IN CALLED SUBPROGRAM --SUR103--, YET ITS COMMON BLOCK  
----\*RLK\*--- IS NOT AVAILABLE TO CALLING SUBPROGRAM --SYSMAIN--.  
HENCE, A COMPUTED VALUE WILL BE LOST. (SEE NOTE 1)

\*\* 216 \*\* COMMON VARIABLE ----\*C\*---- IS ASSIGNED A VALUE ON ALL PATHS  
IN CALLED SUBPROGRAM --SUR215--, YET ITS COMMON BLOCK  
----\*RLK\*--- IS NOT AVAILABLE TO CALLING SUBPROGRAM --SYSMAIN--.  
HENCE, A COMPUTED VALUE WILL BE LOST. (SEE NOTE 1)

\*\* 217 \*\* COMMON VARIABLE ----\*C\*---- IS ASSIGNED A VALUE ON SOME PATHS  
IN CALLED SUBPROGRAM --SUR103--, YET ITS COMMON BLOCK  
----\*RLK\*--- IS NOT AVAILABLE TO CALLING SUBPROGRAM  
--SYSMAIN--. HENCE, A COMPUTED VALUE MAY BE LOST. (SEE  
NOTE 1)

- \*\* 217 \*\*** COMMON VARIABLE ----\*D\*---- IS ASSIGNED A VALUE ON SOME PATHS  
 IN CALLED SUBPROGRAM --\*SUN215\*--, YET ITS COMMON BLOCK  
 ----\*PLK\*---- IS NOT AVAILABLE TO CALLING SUBPROGRAM  
 --\*SYSMAIN\*--, HENCE, A COMPUTED VALUE MAY BE LOST. (SEE  
 NOTE 1)
- \*\* 218 \*\*** COMMON VARIABLE ----\*T\*---- IS INITIALIZED IN BLOCK DATA.  
 IT IS ASSIGNED A VALUE ON ALL PATHS IN CALLED SUBPROGRAM  
 --\*SUN215\*--, YET ITS COMMON BLOCK ----\*IBD\*---- IS NOT AVAILABLE  
 TO CALLING SUBPROGRAM --\*SYSMAIN\*--, HENCE, UNDEFINITION WILL  
 OCCUR UPON EXIT FROM --\*SUN215\*--. (SEE NOTE 2)
- \*\* 219 \*\*** COMMON VARIABLE ----\*W\*---- IS INITIALIZED IN BLOCK DATA.  
 IT IS ASSIGNED A VALUE ON SOME PATHS IN CALLED SUBPROGRAM  
 --\*SUN215\*--, YET ITS COMMON BLOCK ----\*IBD\*---- IS NOT AVAILABLE  
 TO CALLING SUBPROGRAM --\*SYSMAIN\*--, HENCE, UNDEFINITION MAY  
 OCCUR UPON EXIT FROM --\*SUN215\*--. (SEE NOTE 2)
- \*\* 226 \*\*** IN THE MAIN PROGRAM, COMMON VARIABLE ----\*CA\*---- IS  
 ASSIGNED A VALUE IN BLOCK NO. 32 AND IS EITHER  
 ASSIGNED A VALUE THEREAFTER BEFORE BEING REFERENCED,  
 OR IS NOT SUBSEQUENTLY REFERENCED, ON ALL PATHS.  
 ONE SUCH PATH, INDICATED BY BLOCK NUMBERS, IS  
 32 33
- \*\* 227 \*\*** IN THE MAIN PROGRAM, COMMON VARIABLE ----\*HA\*---- IS  
 ASSIGNED A VALUE IN BLOCK NO. 34 AND IS EITHER  
 ASSIGNED A VALUE THEREAFTER BEFORE BEING REFERENCED,  
 OR IS NOT SUBSEQUENTLY REFERENCED, ON SOME PATHS.  
 ONE SUCH PATH, INDICATED BY BLOCK NUMBERS, IS  
 34 35 37 - 48
- \*\* 228 \*\*** IN THE MAIN PROGRAM, AN ELEMENT OF THE COMMON ARRAY  
 ----\*Y22A\*-- IS ASSIGNED A VALUE IN BLOCK NO. 37  
 AND THE ARRAY IS NOT SUBSEQUENTLY REFERENCED ON ANY PATH.
- \*\* 229 \*\*** LOCAL VARIABLE ----\*I23A\*-- IS ASSIGNED A VALUE IN BLOCK  
 NO. 5 AND IS EITHER ASSIGNED A VALUE THEREAFTER BEFORE  
 BEING REFERENCED, OR IS NOT SUBSEQUENTLY REFERENCED,  
 ON ALL PATHS.  
 ONE SUCH PATH, INDICATED BY BLOCK NUMBERS, IS  
 5 - 48
- \*\* 229 \*\*** LOCAL VARIABLE ----\*X\*---- IS ASSIGNED A VALUE IN BLOCK  
 NO. 17 AND IS EITHER ASSIGNED A VALUE THEREAFTER BEFORE  
 BEING REFERENCED, OR IS NOT SUBSEQUENTLY REFERENCED,  
 ON ALL PATHS.  
 ONE SUCH PATH, INDICATED BY BLOCK NUMBERS, IS  
 17 18
- \*\* 229 \*\*** LOCAL VARIABLE ----\*K\*---- IS ASSIGNED A VALUE IN BLOCK  
 NO. 26 AND IS EITHER ASSIGNED A VALUE THEREAFTER BEFORE  
 BEING REFERENCED, OR IS NOT SUBSEQUENTLY REFERENCED.

ON ALL PATHS.

ONE SUCH PATH, INDICATED BY BLOCK NUMBERS, IS  
26 27 28

\*\* 230 \*\* LOCAL VARIABLE ---X230--- IS ASSIGNED A VALUE IN BLOCK NO. 34 AND IS EITHER ASSIGNED A VALUE THEREAFTER BEFORE BEING REFERENCED, OR IS NOT SUBSEQUENTLY REFERENCED, ON SOME PATHS. ONE SUCH PATH, INDICATED BY BLOCK NUMBERS, IS  
39 40 41

\*\* 231 \*\* AN ELEMENT OF THE LOCAL ARRAY ---XDAT--- IS ASSIGNED A VALUE IN BLOCK NO. 42 AND THE ARRAY IS NOT SUBSEQUENTLY REFERENCED ON ANY PATH.

\*\* 231 \*\* AN ELEMENT OF THE LOCAL ARRAY ---XDT--- IS ASSIGNED A VALUE IN BLOCK NO. 13 AND THE ARRAY IS NOT SUBSEQUENTLY REFERENCED ON ANY PATH.

\*\* 232 \*\* BLOCK NO. 43  
A POSSIBLE ILLEGAL SIDE EFFECT HAS BEEN DETECTED. IT OCCURS VIA A VARIABLE PASSED IN AN ARGUMENT LIST. THIS VARIABLE HAS APPEARED AT LEAST TWICE IN A STATEMENT -- IN ONE APPEARANCE IT IS USED AS STRICT INPUT AND IN THE OTHER AS STRICT OUTPUT.

|          | CALLING SUBPROGRAM | CALLED SUBPROGRAM |
|----------|--------------------|-------------------|
|          | --SYSMAIN--        | ---b201---        |
| ARGUMENT | -----X-----        | -----X-----       |
| POSITION | 1                  | 1                 |

\*\* 232 \*\* BLOCK NO. 44  
A POSSIBLE ILLEGAL SIDE EFFECT HAS BEEN DETECTED. IT OCCURS VIA A VARIABLE PASSED IN AN ARGUMENT LIST. THIS VARIABLE HAS APPEARED AT LEAST TWICE IN A STATEMENT -- IN ONE APPEARANCE IT IS USED AS STRICT INPUT AND IN THE OTHER AS STRICT OUTPUT.

|          | CALLING SUBPROGRAM | CALLED SUBPROGRAM |
|----------|--------------------|-------------------|
|          | --SYSMAIN--        | ---w201---        |
| ARGUMENT | -----CA-----       | -----X-----       |
| POSITION | 1                  | 1                 |

\*\* 233 \*\* BLOCK NO. 30  
A POSSIBLE ILLEGAL SIDE EFFECT HAS BEEN DETECTED. IT OCCURS VIA A COMMON VARIABLE WHICH HAS BEEN REFERENCED (POSSIBLY INDIRECTLY) AT LEAST TWICE IN A STATEMENT -- IN ONE APPEARANCE IT IS USED AS STRICT INPUT AND IN THE OTHER AS STRICT OUTPUT.

|              | CALLING SUBPROGRAM | CALLED SUBPROGRAM |
|--------------|--------------------|-------------------|
|              | --SYSMAIN--        | ---w201---        |
| VARIABLE     | -----CA-----       | -----CA-----      |
| COMMON BLOCK | ---BLK1---         | ---BLK1---        |

\*\* 233 \*\* BLOCK NO. 44  
A POSSIBLE ILLEGAL SIDE EFFECT HAS BEEN DETECTED. IT OCCURS VIA A COMMON VARIABLE WHICH HAS BEEN REFERENCED (POSSIBLY INDIRECTLY) AT LEAST TWICE IN A STATEMENT -- IN ONE APPEAR-

ANCE IT IS USED AS STRICT INPUT AND IN THE OTHER AS STRICT

OUTPUT.

| VARIABLE     | CALLING SUBPROGRAM | CALLED SUBPROGRAM |
|--------------|--------------------|-------------------|
| COMMON BLOCK | --*SYSMAIN*        | ---*W201*---      |
|              | ----*CA*---        | ----*CA*---       |
|              | ----*BLK1*---      | ----*HLK1*---     |

\*\* 234 \*\* BLOCK NO. 45  
 A POSSIBLE ILLEGAL SIDE EFFECT HAS BEEN DETECTED. IT OCCURS  
 VIA A GLOBAL VARIABLE REFERENCED IN AN ARITHMETIC STATEMENT  
 FUNCTION. THIS VARIABLE HAS APPEARED AT LEAST TWICE IN A  
 STATEMENT -- IN ONE APPEARANCE IT IS USED AS STRICT INPUT AND  
 IN THE OTHER AS STRICT OUTPUT.

| VARIABLE | CALLING SUBPROGRAM | CALLED SUBPROGRAM |
|----------|--------------------|-------------------|
|          | --*SYSMAIN*        | ---*LASHF*---     |
|          | ----*C*---         | ---               |

M E S S A G E S  
 - - - - -

| MESSAGE NUMBER | DESCRIPTION |
|----------------|-------------|
|----------------|-------------|

\*\* 301 \*\* COMMON VARIABLE ---\*D?18\*--- IN BLOCK ---\*BLK1\*--- OF  
 SUBPROGRAM --\*SYSMAIN\*-- IS INITIALIZED IN BLOCK DATA.

\*\* 301 \*\* COMMON VARIABLE ---\*W\*--- IN BLOCK ---\*IRD\*--- OF  
 SUBPROGRAM --\*SUR215\*-- IS INITIALIZED IN BLOCK DATA.

\*\* 303 \*\* THE FOLLOWING DATA FLOW OCCURS THROUGH COMMON WHEN SUBPROGRAM  
 --\*SUR103\*-- IS CALLED.

| COMMON BLOCK | VARIABLE      | INPUT CLASSIFICATION | OUTPUT CLASSIFICATION |
|--------------|---------------|----------------------|-----------------------|
| ---*BLK1*--- | ----*CA*---   | STRICT               | NON                   |
| ---*BLK1*--- | ----*D218*--- | STRICT               | NON                   |
| ---*BLK1*--- | ----*Y228*--- | STRICT               | NON                   |

\*\* 303 \*\* THE FOLLOWING DATA FLOW OCCURS THROUGH COMMON WHEN SUBPROGRAM  
 ---\*W201\*--- IS CALLED.

| COMMON BLOCK | VARIABLE    | INPUT CLASSIFICATION | OUTPUT CLASSIFICATION |
|--------------|-------------|----------------------|-----------------------|
| ---*BLK1*--- | ----*CA*--- | STRICT               | NON                   |

\*\* 304 \*\* I/O CLASSIFICATION OF ARGUMENTS AND COMMON VARIABLES  
 FOR --\*SYSMAIN\*--

COMMON BLOCK ----\*H)----

AVAILABILITY = ORIGINAL

| ARGUMENTS<br>POSITION | NAME          | INPUT CLASS | OUTPUT CLASS |
|-----------------------|---------------|-------------|--------------|
| 1                     | ----*CA1*---- | INPUT       | NON          |
| 2                     | ----*RA*----  | NON         | STRICT       |

COMMON BLOCK ----\*RLK1\*---

AVAILABILITY = ORIGINAL

| ARGUMENTS<br>POSITION | NAME          | INPUT CLASS | OUTPUT CLASS |
|-----------------------|---------------|-------------|--------------|
| 1                     | ----*CA*----  | STRICT      | STRICT       |
| 2                     | ----*D21R*--- | STRICT      | NON          |
| 3                     | ----*Y22H*--- | STRICT      | STRICT       |

! ! ----\*LASHF\*---

| ARGUMENTS<br>POSITION | NAME        | INPUT CLASS | OUTPUT CLASS |
|-----------------------|-------------|-------------|--------------|
| 1                     | ----*X*---- | STRICT      | NON          |
| 2                     | ----*Y*---- | STRICT      | NON          |

NOTES

NOTE 1  
---- -  
ALTHOUGH DETECTED IN THIS SUBPROGRAM, THE CAUSE FOR THIS  
DIAGNOSTIC MAY HAVE OCCURRED AT A DEEPER LEVEL OF SUBPROGRAM  
REFERENCES AND BEEN PROPAGATED UP TO THIS ONE.

NOTE 2  
---- -  
IF MESSAGE 301 CONCERNING THIS VARIABLE APPEARS IN THE  
OUTPUT, IT MAY PROVIDE ADDITIONAL USEFUL INFORMATION  
ABOUT THE DATA FLOW AMONG SUBPROGRAMS.

DISTRIBUTION LIST

| <u>No. of Copies</u> | <u>Organization</u>  | <u>No. of Copies</u> | <u>Organization</u>   |
|----------------------|--|----------------------|---|
| 12                   | Commander<br>Defense Documentation Center<br>ATTN: DDC-TCA<br>Cameron Station<br>Alexandria, VA 22314  | 1                    | Commander<br>US Army Missile Materiel<br>Readiness Command<br>ATTN: DRSMI-AOM<br>Redstone Arsenal, AL 35809   |
| 1                    | Commander<br>US Army Materiel Development<br>and Readiness Command<br>ATTN: DRCDMD-ST<br>5001 Eisenhower Avenue<br>Alexandria, VA 22333        | 1                    | Commander<br>US Army Tank Automotive<br>Research & Development Cmd<br>ATTN: DRDTA-UL<br>Warren, MI 48090  |
| 1                    | Commander<br>US Army Aviation Research<br>and Development Command<br>ATTN: DRSAV-E<br>12th and Spruce Streets<br>St. Louis, MO 63166           | 1                    | Commander<br>US Army Armament Materiel<br>Readiness Command<br>ATTN: DR SAR-LEP-L, Tech Lib<br>Rock Island, IL 61299  |
| 1                    | Director<br>US Army Air Mobility Research<br>and Development Laboratory<br>Ames Research Center<br>Moffett Field, CA 94035                     | 4                    | Commander<br>US Army Armament Research<br>and Development Command<br>ATTN: DRDAR-TSS (2 cys)<br>DRDAR-MS, D. Grobstein<br>DRDAR-MSE, S.A. Goldberg<br>Dover, NJ 07801 |
| 1                    | Commander<br>US Army Electronics Research<br>and Development Command<br>Technical Support Activity<br>ATTN: DELSD-L<br>Fort Monmouth, NJ 07703 | 1                    | Director<br>US Army TRADOC Systems<br>Analysis Activity<br>ATTN: ATAA-SL, Tech Lib<br>White Sands Missile Range<br>NM 88002   |
| 1                    | Commander<br>US Army Communications Rsch<br>and Development Command<br>ATTN: DRDCO-SGS<br>Fort Monmouth, NJ 07703                              | 2                    | Commander<br>US Army BMD Advanced<br>Technology Center<br>ATTN: ATC-P, Mr. C. Vick<br>Dr. Carl Davis<br>P. O. Box 1500<br>Huntsville, AL 35807                        |
| 1                    | Commander<br>US Army Missile Research<br>and Development Command<br>ATTN: DRDMI-R<br>Redstone Arsenal, AL 35809                                | 1                    | AFSC/ASD (Dr. Richard Sylvester)<br>Wright-Patterson AFB, OH 45433  |

DISTRIBUTION LIST

| <u>No. of<br/>Copies</u> | <u>Organization</u>   | <u>No. of<br/>Copies</u> | <u>Organization</u>  |
|--------------------------|---|--------------------------|--|
| 1                        | Director<br>National Aeronautics and<br>Space Administration<br>Langley Research Center<br>ATTN: Dr. T. Straeter<br>Mail Stop 125A<br>Hampton, VA 23665 | 1                        | The Johns Hopkins University<br>ATTN: Dr. John H. Manley<br>Baltimore, MD 21234  |
| 2                        | Boeing Computer Science Co.<br>ATTN: Dr. L. Stucki<br>Mr. J. Brown<br>Seattle, WA 98124   | 1                        | University of Maryland at<br>College Park<br>Dept of Computer Science<br>ATTN: Dr. V. R. Basili<br>Dr. M. V. Zelkowitz<br>College Park, MD 20742 |
| 1                        | Flow General Corporation<br>ATTN: Sabina Saib<br>P. O. Box 3587<br>Santa Barbara, CA 93105  | 1                        | University of Massachusetts<br>Dept of Computer and<br>Information Science<br>ATTN: Dr. L. Clarke<br>Amherst, MA 01002                           |
| 1                        | SOFTECH, Inc.<br>ATTN: Dr. J. Goodenough<br>460 Totten Pond Road<br>Waltham, MA 02154   | 1                        | University of North Carolina<br>at Chapel Hill<br>Department of Computer Science<br>ATTN: Dr. D. Parnas<br>Chapel Hill, NC 27514                 |
| 1                        | Software Research Associates<br>ATTN: Dr. E. F. Miller, Jr.<br>P. O. Box 2432<br>San Francisco, CA 94126  | 1                        | University of Texas at Austin<br>Department of Computer Science<br>and Computation Center<br>ATTN: Dr. J. C. Browne<br>Austin, TX 78712          |
| 2                        | SRI International<br>ATTN: Dr. B. Elspas<br>Dr. J. Goldberg<br>333 Ravenswood Avenue<br>Menlo Park, CA 94025  | 1                        | Virginia Polytechnic and<br>State University<br>Department of Computer Science<br>ATTN: B. G. Claybrook<br>Blacksburg, VA 24061                  |
| 2                        | University of Colorado<br>Dept of Computer Science<br>ATTN: Dr. L. Fosdick<br>Dr. L. Osterweil<br>Boulder, CO 80309                                     |                          | <u>Aberdeen Proving Ground</u><br>Cdr, USATECOM<br>ATTN: Mr. Barnhart<br>DRSTE-SG-H  |
| 2                        | University of Delaware<br>ATTN: Dr. H. Khalil<br>Dr. L. Levy<br>Newark, DE 19711  |                          | (continued on next page)   |

DISTRIBUTION LIST

Aberdeen Proving Ground (continued)

Dir, USAMSA

ATTN: Dr. Sperrazza  
Dr. Atzinger  
Mr. L. Bain  
Mr. J. Bannon  
Mr. E. Belbot  
MAJ Cecil  
Mr. Clay  
Dr. Cooper  
Dr. Cummings  
Mr. Daily  
Mr. Dombeck  
Dr. Dubin  
Mr. Holloway  
Mr. Iten  
Mr. Kilminster  
Mr. Lufkin  
Mr. McCarthy  
Mr. Sebra  
Dr. Streilein  
Mr. Towson  
Mr. Warfield